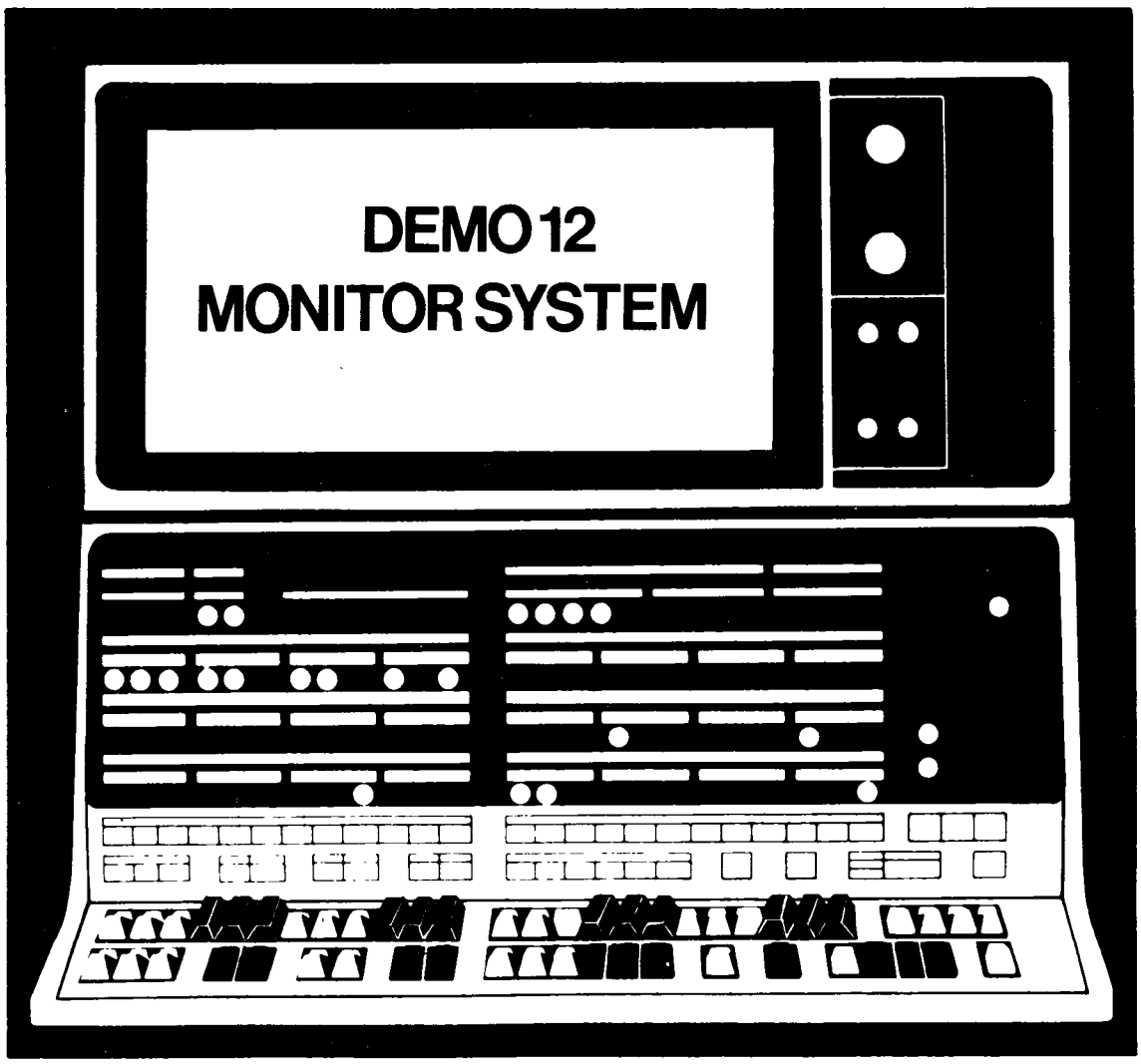
pdp12

TECHNICAL REFERENCE MANUAL



digital equipment corporation·maynard massachusetts

DEMO12

MONITOR SYSTEM

TECHNICAL REFERENCE MANUAL

For additional copies of this document, order No. DEC-12-MRZA-D from Program Library, Digital Equipment Corp., Maynard, Mass. 01754 Price $1.00.

DIGITAL EQUIPMENT CORPORATION • MAYNARD, MASSACHUSETTS

1st Printing April 1970
DEC-12-MRZA-D

Your attention is invited to the last two pages of this manual. The Reader's Comments page, when filled in and returned, is beneficial to both you and DEC. All comments received are considered when documenting subsequent manuals, and when assistance is required, a knowledgeable DEC representative will contact you. The Software Information page offers you a means of keeping up-to-date with DEC's software.

Instruction times, operating speeds and the like are included in this manual for reference only; they are not to be taken as specifications.

PDP-12 DEMO USER PROGRAM

DISCLAIMER

The accompanying LINCtape entitled LAP6-DIAL* and DEMO Monitor System contains a variety of data acquisition,reduction, manipulation, and presentation programs which operate on the PDP-12A. All binaries may be loaded via the DEMO Monitor or with LAP6-DIAL, and are either self-starting or are started by pressing I/O PRESET and START 20. Abstracts describing the operation of each of these programs are attached.

All credit for these programs belongs, in most cases, to the original authors. The only contribution of the current authors was to convert these programs to run under the DEMO Monitor as a convenience to PDP-12 users.

The question of responsibility for these programs is a difficult one. Obviously the original authors cannot be held responsible for correct operation of the modified programs in a new machine. Nor can Digital Equipment Corporation claim to be responsible, as we do not have adequate descriptions of the internal structure of some of these programs. All appear to work. We would suggest that the programs be checked out before basing any substantial conclusions on results obtained with them. LAP6-DIAL sources have been obtained for most data reduction programs to allow program verification and modification.

* Hereafter referred to as DIAL for brevity

CONCISE CONTENTS

APPENDICES

TABLE OF CONTENTS

1. INTRODUCTION

This manual contains detailed information on the PDP-12 DEMO12 Monitor System (also referred to as DEMO12).

The first 8 sections of this manual include information on file structures, input/output, core usage, states of the DEMO Monitor, transitions from state to state, and the important flags and indicators.

DEMO12 consists of five main programs, LOADER, INITLIZE, SEQØ, SEG1, and BALLON, which are discussed in sections 9, 10, 11, and 12. Refer to Section 16 for the procedure to combine these programs.

Section 13 outlines the Monitor messages that appear on the scope.

Section 14 describes how to add a program so it can be controlled by DEMO12.

The procedure for changing the Summary and Help Frames is briefly described; a more detailed description of this procedure is in the DEMO12 Users' Guide, DEC-12-UXZA-D.

Operating Procedures are in Section 17. DEMO12 listings are available from the Program Library.

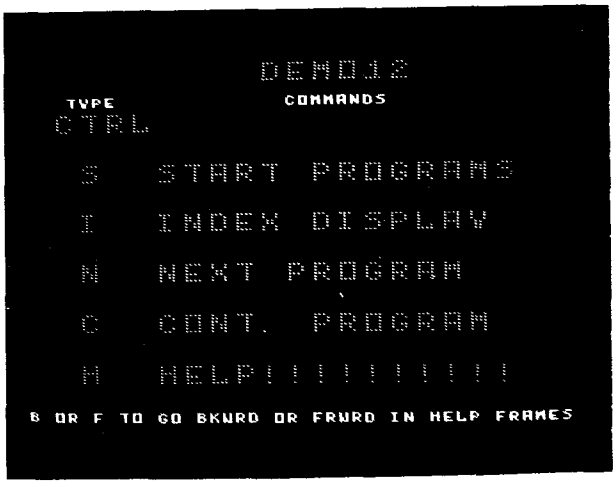
2.0 Hardware Requirements

The monitor will run on a basic PDP-12A. Use is made of the VR12 display, the LINCtape control and two units, the console Teletype [1] and 2K of the basic 4K memory. Options not included in the PDP-12A basic system are not required, but up to 6K of core may be used by the DEMO User Programs (DUPs). The DUP loader will load 6K programs.

The monitor includes a provision for the handling of interrupts from devices which are not part of the basic system. (See paragraph 8.1 for a description of this provision.)

2.1 Software Environment

The monitor is entirely self-contained and requires no additional software for the operation of the system, except DUPs which must conform to certain core restraints and not use certain instructions directly. (See paragraph 14.0)



[1] Teletype is a registered trademark of Teletype Corporation.

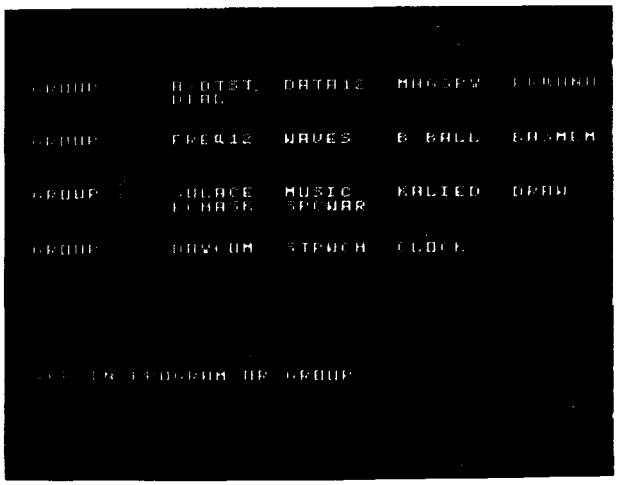
3.0 Special Index Considerations

The DEMO System contains the DEMO Monitor in blocks Ø-17. Block 20 is a duplicate of the DIAL Index, block 346, and is located in block 20 so that the DEMO Monitor needn't go all the way up to 346 to get DIAL's Index, thus saving significant amounts of tape shuffling time. DIAL and the DEMO User Programs (DUPs) are in the rest of the tape, along with some Utility programs and sources.

A special entry in the DIAL Index must reserve blocks 0-20, 103-152 for the DEMO. Although no special format is required for this reservation, the method suggested is an entry in the DIAL Index as indicated in paragraph 3.2.

3.1 The DIAL/DEMO Index

Never use a period at the beginning of a file name unless it is a DEMO User Program.



3.1 DIAL/DEMO Tape Storage Allocation

REQUIRED FOR DEMO MONITOR SYSTEM		REQUIRED FOR DIAL SYSTEM		NONE OF THESE FILES REQUIRED FOR DIAL OR DEMO USE. DIAL USER FILES DELETABLE	
BLOCK	FILETYPE	BLOCK	FILETYPE	BLOCK	FILETYPE
0	DEMO MONITOR	270	DIAL FREE	470	DIAL USER FILES
23		277			
24	BINARY SUMMARY AND HELP FRAMES FOR ALL DUPS	300	DIAL SYSTEM		
102		345			
103	DEMO SCRATCH AREA, 50 BLKS	346	DIAL INDEX		
152		347			
153	BINARIES OF EXECUTABLE DEMO USER PROGRAMS (DUPS)	350	MORE DIAL SYSTEM	634	MORE DIAL USER FILES
267		367		635	DIAL USER FILE SOURCES OF DEMO MONITOR AND ASSEMBLED BINARIES OF DEMO MONITOR SUBPARTS
		370	DIAL EDITOR SOURCE WORK AREA 100 BLOCKS		
		467		777	

NAME	SOURCE BN	SOURCE BLKS	BINARY BN	BINARY BLKS	
LOADER.			0	1	DIAL UTILITY & MISC. USER FILE
SEG01-3.			1	3	
SEG11-4.	DEMO MONITOR SYSTEM PROPER		4	4	
INITLIZ.			10	1	
GREETIN.			11	1	
BALLOON.			12	5	
SEG04-4.			17	1	
DMOINDX.			20	2	
Q AND A.			22	2	
.DA-DTST	24	2	165	3	DEMO MONITOR BINARY & SOURCES
.DDATA12	26	4	37	6	
.DMAGSPY	32	2	153	10	
.DFRQANA	35	2	72	11	
.DDIAL	45	1	163	2	
.EFREQ12	46	2	170	3	
.EWAVES	50	2	173	3	
.EB.BALL	52	1	176	3	
.EBASMEM	53	1	201	5	
.FSOLACE	54	1	206	2	
.FMUSIC	55	1	210	6	
.FKALEID	56	1	216	2	
.FDRAW	57	2	220	3	
.FECHASK	61	2	223	3	
.FSPCWAR	63	3	226	16	
.TDAYCOM	66	1	244	4	
.TSTPWCH	67	2	250	6	
.TCLOCK	71	1	256	5	
SCRATCH.	103	50			
GREETING			776	1	
INDEXSRC	263	5			
DIAL-V2	522	2			
MARK12			511	7	
PIP			470	21	
EX.PROG.	524	1	520	2	
AD DEMO	526	1	614	3	
BINLOAD			535	3	
L8SIM			540	3	
CONVERT			543	5	
FOCAL8K			550	23	
FOCAL4K			573	21	
CAROLS			617	11	
LOADER	635	2	637	2	
INITLIZE	641	17	660	3	
SEG0	663	34	717	6	
SEG1	725	36	763	6	
BALLOON			771	5	
04-01-70	527	2			

DEMO USER PROGRAMS SOURCE AREA CONTAINS HELP FRAMES AND THE BINARY AREA CONTAINS THE ACTUAL PROGRAM

4.0 Files

4.1 DIAL Directory

The DIAL directory is assumed to be as specified: each entry is eight words consisting of eight stripped 6-bit ASCII characters and four words of block information, with all unused words filled with 5757_8. It is assumed to be on blocks 346 and 347 of the DEMO tape. The DEMO Monitor doesn't use 346 or 347 directly, but rather a copy of the DIAL directory, block 346, copied into block 20 of the DEMO tape using PIP. Note that a change to the DIAL directory then will not affect the DEMO INDEX unless block 346 is again modified.

4.2 DEMO Index

The DEMO Index is created from the DIAL Index copy in block 20 during initialization and is written on tape block 17. The last 57_8 words of the directory are part of the instruction trap handling routines.

Each entry in the DIAL Index for which the first character is 56_8 ('.') is copied into the DEMO Index. Only the first 30 entries with '.' as the first character are so copied -- the rest are ignored. The entries in the DEMO Index are then sorted by the second character of the entry.

Filler characters (77_8) in the name portion of each entry are converted to spaces.

4.3 DEMO Index File

The DEMO Index File is a display buffer which is created during initialization and written on tape block 16. It consists entirely of display data in a special format peculiar to the DEMO Monitor, see paragraph 11.2.

5.0 Input/Output

The DEMO Monitor is cognizant of only the console Teletype and the LINCtape controller when it handles interrupts unless the DUP requested an interrupt chain extension. See paragraph 14.8.4. All input to the monitor normally comes from the Teletype or from the tape controller. All output normally goes to the Teletype, the VR12 display, or LINCtape. A-D input comes via the DUP.

The use of tape units is important. In general, tape 1, if available, will always be in the vicinity of blocks 16 and 17, the display index and index respectively. All DUPs are loaded from tape Ø.

5.1 Other I/O

I/O with other devices can only be handled by a DUP and if that device generates interrupts, the interrupt chain extension feature will have to be used.

6.0 Core Layout/Tape Blocks

6.1 Initialization Time

CORE		Tape Block
0 - 377	Interrupt and trap handler	1
400 - 777	Interrupt and trap handler	2
1000-1377	Interrupt and trap handler	3
1400-1777	Not used	
2000-2377	Monitor State routines,	4
2400-2777	subroutines, and	5
3000-3377	messages	6-7
4000-4377	Initialization (INITLIZ.)	10
4400-4777	Initialization (GREETIN.)	11
5000-5377	DIAL Index (copied from block 20)	20
5400-5777	Not used	
6000-6377	First half of	12
6400-6777	Talk Balloon	13
7000-7377	DEMO Index (during sort) / Second Half of (after sort)	11/14
7400-7777	Index Display / Talk Balloon	17/15

6.2 Runtime

0 - 377	Interrupt and trap handler	1
400 - 777	Interrupt and trap handler	2
1000-1377	Interrupt and trap handler	3
1400-1777	Tape Buffer-Index, display index, summary or Help	
2000-2377	state routines, messages	4
2400-2777	state routines, messages	5
3000-3377	Subroutines	6
3400-3777	Subroutines	7
4000-17777	Demo User Program (DUP)	User specified

7.0 Monitor State System

Changes of Monitor status when a control key is struck are handled through a state change table in the Monitor. The table contains entries for each state and each control character such that the next state can be determined by existing state and control characters.

7.1 States

The following are the states of the DEMO Monitor which can be in effect.

1. Error Display or Monitor Help.
2. Display Index and accept as input the name of a DUP.
3. Load DUP Summary frame.
4. Load a DUP.
5. Display the DUP Summary frame and wait for Control S.
6. Set state 5 to not wait for Control S and go to state 4.
7. Execute the DUP(using the saved state if one exists).
8. Display DUP Help frame
9. Display DUP Help (Three states because of differing state transitions)
10. Display DUP Help
11. Select next DUP in same group and load it.
12. If a state is saved, restart the current DUP at the saved state.
13. If the DUP is loaded, start it.
14. If a DUP is selected, load it.

7.2 Transition Tables

There are separate transition tables for each control character. The new state is a half-word entry in the table.

Note that table entries are in octal, but refer to decimal state numbers. This can lead to some confusion, so care should be taken in this area.

7.2.1 Control S - Start the DUP

The transition table for Control S is tagged TS in bank Ø. Its entries are:

Existing Entries	New State
1	14
2	13
3	3
4	6
5	7
6	6
7	6
8	6
9	7
10	6
11	11
12	12
13	13
14	14

7.2.2 Control I - Go to DEMO Monitor Index Display

Since every entry in the table would be a 2, no table is required. In place of a table address, control I's entry is a -2 which is properly interpreted as a transition to state 2.

7.2.3 Control N - Load the next DUP

As in 7.2.2, no table is required. The entry for Control N is a -13_8 or new state 11_{10}.

7.2.4 Control C - Continue Current Program

The transition table for Control C is tagged TC in block Ø. Its entries are:

Existing State	New State
1	12
2	12
3	3
4	4
5	5
6	6
7	7
8	6
9	7
10	7
11	11
12	12
13	13
14	14

Note: A zero entry indicates that no state transition is to take place.

7.2.5 Control H - Help

The table for Control H is tagged TH in bank Ø. Its entries are as follows:

Existing State	New State
1	1
2	1
3	1
4	8
5	9
6	8
7	10
8	1
9	1
10	1
11	1
12	1
13	1
14	1

Note: Any transition to state 1 will cause "MONITOR HELP" to be displayed. Error displays are handled in a special manner.

8.0 Flags and Indicators

The following flags and indicators may be of interest for debugging or adding any changes to the Monitor.

NAME	BANK	LENGTH	TYPE	SPECIAL VALUES
INEXT	0	1	8 mode address	Ø = No extension
STSUD	0	1	Flag	Ø = No state saved 7 = State saved
KSTF	0	1	Flag	Ø = No keyboard character available. 1 = Char.available in ICHR.
ICHR	0	1	Data	last char. rec'd.(ASCII)
TFLG	0	1	Flag	0 = Printer not busy 1 = Printer busy
FOOSWT	0	1	L.Inst.	Contains a jump to one of several appropriate routines to handle tape interrupts.
TAPINT	Same as FOOSWT			
ISTATE	0	10_8		Interrupt state
TSTATE	0	10_8	diagram below	Trap state
DSTATE	0	10_8		Saved DEMO state

1. overflow state (SVFLO)
2. AXO register (SVAXO)
3. SF register (SVEFS)
4. MQ (SVMQ)
5. Fields (SVFLD)
6. PC (SVPC)
7. Link & Mode (SVLNK)
10. AC (SVAC)

THISDP	1	1	$L_{addr.}$	Pointer to DEMO INDEX entry for current DUP.
STATE	1	1	Data	Current state number
ZDML	1	1	Flag	0-DUP is not loaded 1-DUP is loaded
TAPMES	1	1	$L_{addr.}$	Location of message to be displayed during tape op.

8.1 INEXT (used by OPR11)

The 12-bit PDP-8 address in INEXT is placed there by an OPR11. The call is:

```
 OPR 11
PMODE
 ROUTINE NAME (a 12-bit address of the interrupt service
LMODE          routine located in the current DUP using OPR11).
```

or any equivalent sequence. (See paragraph 14.8.4 and 11.1.4)

Once specified, the extended interrupt routine will be executed for each interrupt from a device other than LINCtape, keyboard, or TTY printer. The Monitor fields and handles all interrupts from those devices.

Obviously, the above code sequence can only be executed from within the DUP. However, the extension routine remains in effect when DUP execution is suspended or terminated by any means.

The extension routine will be retracted only when a new DUP is about to be loaded, the same DUP is being reloaded, or the following code sequence is executed. Care should be exercised in user programs so as not to leave OPR11 on.

```
.
.
.
OPR11
  Ø
.
.
.
```

8.2 STSVD

STSVD is a flag which indicates whether DSTATE contains a valid DUP state or not.

If (STSVD) = Ø, DSTATE does not contain a valid state.

If (STSVD) = 7, DSTATE is a valid state.

No other values will occur.

8.2 STSVD (continued)

Whenever a trapped instruction occurs, or whenever a state change is dictated by a control character, DSTATE is set to the proper values and STSVD is set to 7 by the subroutine COPY. DSTATE thus represents the state to which control should be given to continue DUP execution where it was interrupted.

In the case of trapped instructions, the PC value in DSTATE always indicates the address of the actual instruction which is trapped. Therefore, that instruction will be re-executed if DUP execution is suspended during the trap process and later resumed.

8.3 KSTF, ICHR

KSTF is the hidden keyboard flag. Since the monitor runs with interrupt enabled, the actual keyboard flag will cause an interrupt and will be cleared behind the back of KST instructions. Hence, KST and conventional keyboard IOT instructions are not useful in this environment. Your DUP will never get a chance to check them.

Whenever the keyboard flag goes up, the character is read from the keyboard buffer and stored in ICHR and KSTF is set to 1. KSTF can be tested with the instruction OKST which operates just as KST but tests KSTF instead of the real keyboard flag. The flag is reset to Ø whenever the character which caused it to be set is delivered to the program (see KBD and KBDA instructions in paragraphs 11.2.1.6 and 11.2.1.8).

8.4 TFLG

TFLG is a busy flag for the TTY printer. It is set to 1 whenever a character is sent to the TTY by the Monitor, and is reset when an interrupt indicates that the printer has finished.

8.5 FOOSWT (also referred to as TAPINT)

This location is actually the first word of the routine which is executed when a tape interrupt occurs. It contains a LINC-mode JMP to an appropriate portion of the tape interrupt processor. A further discussion of this switch will be found in section 11.2.2.1.

8.6 ISTATE

ISTATE is a group of 10_8 locations used to preserve the state of the machine when an interrupt occurs.

8.6.1 SVFLO

SVFLO is the first word of ISTATE. If overflow was set when the interrupt occurred, (SVFLO) = 7776 (-1), otherwise it is Ø.

8.6.2 SVAXO

SVAXO contains the tape extended operations register which was in effect at the time of the interrupt.

8.6.3 SVEFS

SVEFS contains the special functions register which was in effect at the time of the interrupt.

8.6.4 SVMQ

SVMQ contains the full 12 bits of the MQ at the time of the interrupt. The MQ is retrieved by the following code:

```
    .
    .
QAC
ROL 1
QLZ
ADD ONE
    .
    .
    .
```

8.6.5 SVFLD

SVFLD contains the fields in effect before the interrupt occurred. SVFLD is always saved in LINC mode via the IOB; RIB instruction.

8.6.6 SVPC

SVPC contains either a PDP-8 mode address or a LINC-mode JMP to the place where the interrupt occurred, depending upon the mode at the time of the interrupt.

8.6.7 SVLNK

SVLNK contains only 2 significant bits. Bit 0 is the state of the link at the time of the interrupt and bit 1 is the mode if set.

8.6.8 SVAC

SVAC contains the contents of the AC register at the time of the interrupt.

8.7 TSTATE

TSTATE is similar to ISTATE, but it indicates the state at the time a trapped instruction occurred.

8.7.1 ISVPC

ISVPC contains a LINC-mode JMP to the command which caused the trap rather than the next instruction in sequence. It is incremented just prior to returning to the state from which the trap occurred.

8.7.2 Interrupt And Trap Returns

Traps cannot occur in PDP-8 mode, but all flags are properly set so that common returns from trap and interrupt are possible.

8.8 DSTATE

DSTATE is a copy of either ISTATE or TSTATE. The copy is only made if

1. a trapped instruction is being executed, or
2. a control character which causes a state change has been received, and
3. DSTATE does not already contain a saved state (see STSVD, paragraph 8.2.)

8.9 THISDP

THISDP is a pointer to the DEMO INDEX entry for the currently loaded DUP. When the Index is in core it is in quarter 3 of segment Ø, therefore, THISDP contains a value between 3400 and 3777. It is always referenced from segment 1 (DF=Ø).

8.10 STATE

STATE contains a unique number for each of the various Monitor routines. It is required in order to determine which routine should be entered when a control character is struck. To illustrate the problem, it should be noted that Control H can cause either DEMO Help or Monitor Help to be displayed, depending upon whether it has been struck once before or not. (This is a simplification. The actual state transitions for each control character are described in section 7.) Since Monitor Help and DEMO Help are displayed by different routines, some means of knowing what is currently going on is required.

STATE always contains a value between 1 and 16_8. The meaning of each value is described in section 7.

8.11 ZDML

ZDML is a flag which indicates whether the DUP is loaded and is a "clean" copy or not. It is set to 1 when the DUP is loaded and is reset to Ø when a new DUP is selected or when the DUP execution is begun.

8.12 TAPMES

TAPMES is a word which contains the address of a message to be displayed while tape operations proceed under trap processor control. Generally, TAPMES is set to the address of the message "TAPE OPERATION IN PROGRESS."

This feature is not available to the DUP.

9.0 Loader (LOADER)

The loader resides on block Ø of the DEMO tape. Its only function is to load the entire DEMO Monitor and initialization program (INITLIZE).

The Loader first re-reads itself into segment 2, quarter 3 and transfers execution to itself in that location. It then proceeds to load blocks 1 through 7 from tape into segment Ø, quarters Ø, 1, and 2 and segment 1, quarters Ø, 1, 2, and 3.

The next two tape blocks, 1Ø and 11, are loaded into segment 2, quarter Ø and quarter 1. These are the initialization routines. Block 11 is loaded into segment 3, quarter 2 as well as segment 2, quarter 1, since the last several words of block 11 are a part of the index display block which will be generated in segment 3, quarter 2.

The "BALLOON" generation information is contained in tape blocks 12 through 15. Two of these are loaded into segment 3, quarter Ø and quarter 1, leaving the rest until after initialization is complete.

Finally, the index block, 17, is read into segment 3, quarter 3 where the new index will be created. The index block contains a routine used only when the DEMO Index is in core.

10.0 Initialization (INITLIZE)

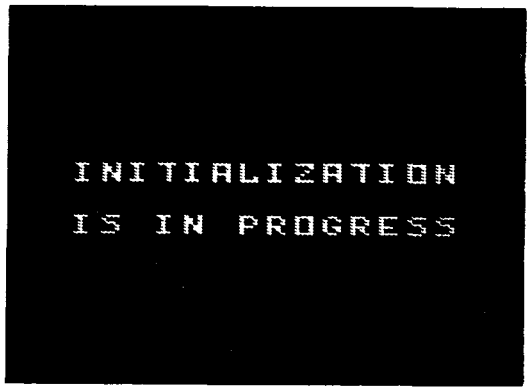
Initialization runs entirely with interrupts off. Its function is to create the DEMO Index and Index display and to write them both on tape, then to pause with the message "TALK TO ME" displayed.

During initialization, all tape operations proceed in no-pause mode with the message "INITIALIZATION IN PROGRESS" displayed for the duration of the tape operation.

First, the duplicate DIAL Index is read into core quarter 2, segment 2. While the tape is positioning, it is constantly observed to assure that progress is being made toward the DIAL Index copy in block 20.

Once the DIAL Index is in, it is scanned for entries which begin with a period (".") and all such entries are copied into the resident index. A maximum of 30 entries can be copied into the resident index. All filler characters in the DIAL Index (77_8) are converted to trailing spaces in the resident index.

The resident index is then sorted by the second character of each entry, 30 passes are required to assure that the resident index is in proper order.



11.0 Segment Ø Routines (SEGØ)

The routines in segment Ø consist almost entirely of interrupt and trap handling. This narrative is intended to describe the operation of those routines while the code is followed in the listing.

11.1 Interrupt Handling

Interrupt handler entries for PDP-8 mode interrupts (location 1) and LINC mode interrupts (location 41) are both provided. Their paths are only separate for a short time. At each entry, the AC is saved (in SVAC), and the PC is saved from location Ø or 4Ø as appropriate. In LINC mode, the saved PC is set to a JMP before it is saved. Processing continues at INTSAV, with a clear AC if the interrupt occurred in LINC mode, and with at least a high order bit in the AC if the interrupt occurred in PDP-8 mode. This bit is preserved with the link and is used to effect a proper return from interrupt processing. All other status is saved in ISTATE by the subroutine SVSUB1. The interrupt processor then proceeds to check flags and execute the appropriate routine.

11.1.1 Tape Interrupts

If tape interrupts are enabled and if the flag is detected with a STD, the interrupt has come from the tape control. TAPINT is the beginning of the processing routine. A further discussion of tape interrupts will be found in paragraph 11.2.

11.1.2 Keyboard Interrupts

If the keyboard flag is up, the keyboard interrupt processor is executed. The character is read from the keyboard buffer and the keyboard flag is cleared. Note that no check is performed to see if the previous character has been delivered via KBD or KBDA operates, so that the new character overlays the old character and the old character is lost.

An internal flag, KSTF, is set to 1 to indicate that a new character is in.

The new character is then matched to a table, KCTAB, to see if it matches any entry. The last entry of the table is the character itself, so a match is bound to occur.

Alternate words of the table indicates what state change is required as follows:

1. Positive entry - the address of a state transition table, shifted left 1 bit and decremented by 1.
2. Negative entry - the next state number.
3. Negative zero entry - no state change required.

In the event that a state change is indicated, the flag KSTF is cleared. The subroutine COPY is called to save the DUP state if it has not already been saved and if, in fact, the DUP was executing when the interrupt occurred. Control is given to the state changer (segment 1). See paragraph 12.9 for further processing.

11.1.3 Teleprinter Interrupts

If the interrupt was neither from tape nor keyboard, the teleprinter flag is tested and, if it is up, it is cleared and the flag TFLG is cleared.

11.1.4 Other Interrupts

If none of the above flags is found, the interrupt extension routine, if any, is executed. If none is present, or if the interrupt extension routine does not find any flags, processing is transferred to ISERØ. Otherwise control is returned via RET8 (see paragraph 11.1.5).

At ISERØ, a power clear is given and a test is made to see if the interrupt is still present. If it is not, the message "SPURIOUS INTERRUPT" is displayed. Execution may be resumed with Control C.

If the interrupt did not clear with power clear, the message "UNCLEARABLE INTERRUPT" is displayed. No return is possible and the DEMO Monitor will have to be restarted.

11.1.5 Interrupt Returns

Return from interrupt processing is through a routine called GO via RET or RET8. Each status is restored and interrupts are turned on before returning to the appropriate place in the appropriate mode.

11.2 Trap Processor Operation (See paragraph 14.8).

The trap processor is entered at location 14Ø in segment Ø. The status is saved in a manner similar to that found in the interrupt processor, except that $1Ø_8$ registers at TSTATE are used instead of those at ISTATE.

Several instructions within the trap processor are modified as part of trap entry as follows.

11.2 Trap Processor (continued)

Tag	Contents changed to	
RP5	LIF n	n was the IF at trap time
RP5+1	LDF m	m was the DF at trap time
IT1	LDF n	
RP2	LDF n	
RP5+2	JMP y	y is the address of the trapped instruction
LINS		The trapped instruction
LINS 1		The next word
RPØ		The address of the second word after the trapped instruction (with two high order bits set).
SVP2		The second word after the trapped instruction
SVP3		The third word

Note also that the PC in the saved trap state points to the instruction which caused the trap rather than the next instruction in sequence.

The subroutine COPY is always called when a trapped instruction is encountered to save the state of the DUP. It is always possible to be interrupted out of the trap processor, but never to be trapped out of the interrupt processor. Hence, the DUP state must be saved before interrupts are enabled.

11.2.1 Operate Instruction Traps

It is now determined whether the trapped instruction is a tape instruction or an operate. If it is an operate, a JMP to the appropriate routine via a JMP table at IOPRT is generated and after enabling interrupts, the routine is entered. The discussion of tape traps continues at 11.2.2.

11.2.1.1 OPR5 - Retrieve file parameters

The routine for OPR 5 is not resident in segment Ø. It is part of the DEMO INDEX on tape block 16. It is always in core when it is needed, however, because OPR5 is never called within the DEMO Monitor,

11.2.1.1 OPR5 - Retrieve (continued)

and because the DEMO Index is read into core before DUP execution can commence or continue it is available for DUP use.

The routine at IRETV moves the parameter into ZPNAME (see paragraph 12.10) and calls the directory lookup routine which is in segment 1. The call is indirect. If the search is successful, the last four words of the directory entry are copied from the DEMO INDEX into the last four words of the parameter area and the routine exits via XP2 (see paragraph 11.2.3).

11.2.1.2 OPR6 - Next DUP

The routine for OPR6, at INDM, calls the state changer in segment 1 with a new state value of 13_8. (See paragraph 6.7 for a description of that state.)

11.2.1.3 OPR7 - Exit this DUP

The routine for OPR7, at IEDM, calls the state changer with a new state value of 2. (See paragraph 12.2 for a description of the operation of that state.)

11.2.1.4 OPR11 - Extend Interrupt Chain

The routine for OPR11, IXCHN, moves the contents of LINS+1, which is a copy of the word following the trapped instruction, to INEXT. It then returns to the program via XP2 (see 11.2.3).

11.2.1.5 OPR12 - OKST

The routine at IOKST tests the hidden keyboard flag and returns to either the next instruction in sequence or the following instruction via XP1 or XP2, respectively.

11.2.1.6 OPR13 - KBDA

The operation of IKBDA is inextricably related to the routine IKBD, and is described along with IKBD in paragraph 11.2.1.8.

11.2.1.7 OPR14 - TYP

The routine ITYP retrieves the saved AC from TSTATE and, after waiting ror the teleprinter to be free, sends the character to the teleprinter. Return to the DUP is via XP1. (See paragraph 11.2.3)

11.2.1.8 OPR15 - KBD

The operation of IKBD and IKBDA are closely linked. In both cases, if no character has been received from the keyboard, and if the "I" bit in the instruction is off, return is made via XP1 with the saved AC cleared, but if the "I" bit is on, control is retained until a character is received.

The important locations for the keyboard input routines are CASE, LNCODE, and ICHR, the actual full ASCII character received from the keyboard. For the moment it may be assumed that CASE contains a Ø.

Once a character is received from the keyboard, control passes to KBDT where CASE is tested. If CASE is non-zero, then LNCODE contains the character to be returned to the calling program. Since CASE is zero, control is passed to KTRNS.

At KTRNS, the trapped instruction is examined. If it is an OPR13 - KBDA, ICHR is placed in LNCODE, CASE is cleared, the character is echoed and a line feed is also typed if the received character was a carriage return.

11.2.1.8 OPR15 - KBD (continued)

Finally, LNCODE is placed in the saved AC, KSTF is cleared and control is returned via XP1.

If the trapped instruction was OPR15 (KBD) then the input character is taken apart and reassembled until the LINC code for the character is in LNCODE and CASE is set to Ø if no case code is required, or 23_8 if it is required. After the character is echoed, CASE is examined. If it is non-zero, it is placed in the saved AC and control returns via XP1.

On the next call to IKBD, CASE will be non-zero, so it will be cleared, KSTF will be cleared, and LNCODE will be transmitted back to the calling program.

11.2.2 Tape Instructions

If the trapped instruction is a tape instruction, further Monitor initialization is required. In case of error, the "UNIT n" message is set to indicate which unit is being operated on. The unit number may be any octal digit.

The tape trap enable bit in the extended functions register is cleared, but no other bits are disturbed. The flags MTBF and WRTF are cleared.

Depending upon the instruction involved, the flags MTBF or WRTF may be set again for MTB instructions or write instructions respectively. Further, the first and last blocks of WCG instructions or the block for WRC and WRI are checked for the range 103-152 (50 blocks) (see paragraph 13.1.4).

The tape interrupt switch is set for a first interrupt, and the timer is set so that instruction time can begin. Finally, an MTB command is generated and issued to the appropriate tape

unit with maintenance mode and no-pause on. The maintenance commands are used to determine whether the unit is selected and, if necessary, write enabled.

Since the remainder of tape trap processing occurs on an interrupt basis, the display routine in segment 1 is now called. It is first necessary to save location Ø in segment 1 since it may be that the trapped tape instruction was in that segment, and location Ø might be vital. Also, a block count error counter is set to -3 at this time.

11.2.2.1 Tape Interrupt Processing

The remainder of activity in the tape trap handler occurs with tape interrupts. At the first tape interrupt, the switch FOOSWT is found to be a JMP FOOS. At FOOS, the flag MTBF is checked and if the trapped instruction was indeed an MTB, control is immediately relinquished via TPXT. The action at TPXT is described below.

If the instruction is not an MTB, the action on the first interrupt is merely to change FOOSWT to a JMP FOOS1 and to reset the timer for instruction timeout. The MTB instruction generated above is again executed and control is returned to the display in the normal manner for interrupts.

The second interrupt causes a transfer of control to FOOS1. Here, a subroutine, TACTST, is executed to read the TAC (Tape Accumulator) into the AC. The TAC contains the number of blocks the tape is away from the destination block at the present time.

11.2.2.1 Tape Interrupt Processing (continued)

TACTST detects the condition that the TAC is zero, but for this discussion, let us assume that the TAC has a non-zero value. Back at FOOS1, this value is made negative if it is not already negative, and stored in a word tagged OTAC. FOOSWT is set to a JMP FOOS2, the timer is reset, the MTB is again executed, and control is returned in the normal manner for interrupts to the display routine in segment 1.

The third and subsequent interrupts cause a transfer of control to FOOS2. At FOOS2, the subroutine TACTST is again executed, but this time the TAC is made positive, progress was not made during the last tape operation and a tape error is indicated. However, the tape has three chances for this kind of error before the error is detected.

Assuming all is well, the new TAC is now made negative and stored at OTAC. The timer is reset, the instruction re-executed, and control is returned as above.

When TACTST detects that the destination block has been reached, it does not return as a subroutine, but proceeds to execute the actual tape command in place. To execute the command in place, a LIF Ø is placed in the location following the two word tape command and a JMP TWAIT is placed in the following location. Then the command is executed. At TWAIT the two words, which were saved earlier are restored, FOOSWT is set to JMP TPXT, the timer is reset, and control is returned to the display subroutine in segment 1 via normal interrupt procedure.

11.2.2.1 Tape Interrupt Processing (continued)

When the last interrupt finally comes in, the routine at TPXT restores segment 1, location Ø, sets FOOSWT to JMP FOO1 to ignore any further interrupts, and returns via XP2 (see paragraph 11.2.3) rather than via the interrupt procedure.

11.2.3 Trap Returns

The saved PC TSTATE is the address of the actual instruction which caused the trap. All trap returns must increment this address once or twice, depending upon the instruction, before using it to return. This incrementing is done at XP2 for incrementing twice and at XP1 for incrementing once. The code is as follows:

```
XP2,    PDP
        PMODE
        ISZ ISVPC
        LMODE
XP1,    PDP
        PMODE
        ISZ ISVPC
         .
         .
         .
```

Note that the instruction PDP at XP1 is sometimes executed in PDP-8 mode. This is of no consequence because the instruction, when interpreted as a PDP-8 mode instruction, is an "AND 2", and the accumulator contains nothing of value at this time.

Once the saved AC is incremented, the return is much like the return from interrupts, except that STSVD will be cleared if the state is 7.

12.0 Segment 1 Routines (SEG1)

The routines in segment 1 are the state routines and several subroutines for use by the Monitor. This description is intended to be used in conjunction with a listing of the SEG1 routine.

12.1 State 1 Routine - YØ1 (Display Error)

The state 1 routine is used to display all error messages. As such, it is sometimes entered directly from another routine with STATE set to some value other than 1.

The contents of the AC at the time of entry is the number of a message in the table YØ1TAB. This value is saved in YØ1DØ.

If YØ1 is entered from a state transition, the AC will be clear on entry. The message thus invoked is the Monitor Help Frame.

Error numbers greater than 5 are tape error messages. Therefore,it is necessary to display the message "UNIT n" preceeding the display of the numbered message.

12.2 State 2 Routine - YØ2 (Display DEMO Index)

The Display Index routine YØ2 is always entered from the State Changer via a state transition.

After reading in the Display Index, the 4 word variable ZPNAME is initialized and the Index is displayed. The message "TYPE IN PROGRAM OR GROUP" is a part of the Display Index and is always at the same place within the Index block.

If a character is entered from the keyboard, this character is stripped of its two high order bits and placed in ZPNAME in the next available character position. If the entered character is a carriage return, the name constructed in ZPNAME is looked up in the DEMO INDEX,trying each possible group letter until it is found or the possibilities are exhausted. Exhausting the possibilities results in a call to YØ1 with 4 in the AC.

12.2 State 2 Routine - YØ2 (continued)

If the entry is found, THISDP and the variable DEMON (3 words), BKSM, BKNM, and BKSB are set from the DEMO INDEX. Then state 3 is entered via a callto the State Changer.

12.3 State 3 Routine - YØ3 (Get Summary Frame)

The state 3 routine intitializes the Monitor flags INEXT, STSVD, and ZDML. State 5 is set to wait for a Control S from the keyboard, and the first block of the DUP HELP file is read into the buffer area in segment Ø. This first block contains the Summary Frame. When it is read in, TAPMES is set to cause the Summary Frame to be displayed during tape operations, and state 4 is entered via the State Changer.

12.4 State 4 Routine - YØ4 (DUP Loader)

The loader, YØ4, is designed to load DIAL format binary from tape into core. It is incapable of loading into the lower half of the first 4K memory bank, which is the Monitor area, but will load the remainder of up to 8K of memory properly. When the load is completed, State 5 is entered via the State Changer.

12.5 State 5 YØ5 (Display Summary Frame)

The State 5 routine functions in either of two ways, depending upon the state of the switch YØ5. YØ5 contains either a NOP or a JMP YØ5A.

If YØ5 contains a NOP, State 5 simply displays the Summary Frame. When a Control S is typed, the interrupt handler will invoke the State Changer to cause a transition to State 7.

12.5 State 5 YØ5 (Display Summary Frame) (continued)

If YØ5 contains a JMP YØ5A, the Index is read into the buffer area in SEGØ, quarter 3, and State 7 is entered (see paragraph 12.7).

12.6 State 6 Routine - YØ6 (LOADING XXXXXX)

The purpose of this routine is simply to eliminate the Summary display replacing it with the message "LOADING XXXXXX" (where XXXXXX is replaced by the name of the DUP being loaded) and set to YØ5 to a JMP YØ5A. State 4 is then entered via the State Changer.

12.7 State 7 Routine

The State 7 routine is a rather special case. When a transition to state 7 occurs, State 5 is entered at YØ5B with interrupts off. At YØ5B, the variable STATE is set to 5, and the DEMO INDEX is then read into segment Ø, quarter 3. STATE is then set to 7 with interrupt off, TAPMES is set to the standard message "TAPE OPERATION IN PROGRESS" and ADML is cleared because the program will not be a clean copy once the DUP is started.

Finally, YØ7, in segment Ø is invoked, which starts the DUP at the saved state or at a fresh state. The fresh state is defined as:

overflow	clear
tape extended operation register	ØØØØ
extended function register	14ØØ
MQ	ØØØØ
fields	IF = 2, DF = 3
PC	20
mode	LINC
link	Ø
AC	ØØØØ

12.8 State 8, 9, and 1Ø Routine - Y1Ø (DEMO Help)

This state is the DEMO Help State. When entered it is initialized so that the first and last frames are known, then the first frame is read from tape and displayed. The characters F, B, Control F, and Control B cause the routine to advance or back up one frame (but not beyond the ends) and to read the new frame from tape if that is necessary, then to display the new frame.

12.9 State Changer (11, 12, 13, 14)

The State Changer is entered at SCI from the interrupt handler in segment Ø. The new state has already been stored in STATE. The routine simply uses the value of STATE to jump to the appropriate routine through a table.

If a new state is 7, interrupts are not turned on by the State Changer.

The routines for states 11, 12, 13, and 14 are included within the State Changer since they are very short. (See paragraph 7.1)

12.9.1 NXSTAT

NXSTAT is an auxiliary entry to the State Changer. It is always called from segment 1. The new state is in the word following the JMP NXSTAT.

12.10 Directory Lookup Routine

There are actually two directory lookup routines. DLOOK looks up the full 8 character name in ZPNAME, attempting to find a match in the DEMO Index. DLOOF only matches the first two characters of ZPNAME for group lookup. If successful, the directory pointer is left in register 1Ø and the AC.

12.11 Display Subroutine

The display routine is based on QANDA, but is nevertheless quite different. There are two display modes.

12.11.1 Normal Display Mode (QANDA compatible)

In the normal mode, the call to DSPLAY is as follows:

```
JMP   DSPLAY
TEXT ADDRESS
  X/Y Coordinates
(all returns)
```

12.11.1.1 Text Format

The following special characters are recognized in the text:

34_8 - Termination character

43_8 - Carriage return

$Ø6_8$ (F) - When the first character of a line only, causes the line to be displayed full-size.

$1Ø_8$ (H) - When the first character of a line only, H generates half-size characters (for QANDA compatibility).

12.11.1.2 X/Y Coordinates

The X and Y coordinates are generated by multiplying each halfword parameter by $1Ø_8$. If both X and Y are zero, the parameter is ignored; display will continue at the beginning of the last line of the previous display.

12.11.2 Special Mode for Index Display

If the first character of the text is a $ØØ_8$, a special mode is entered. In this mode, the call and the interpretation of X/Y are the same as in normal mode, but the text is specially interpreted as follows.

12.11.2 Special Mode for Index Display (continued)

34_8 (\) Terminator
43_8 (#) Carriage return
74_8 (<) Causes half size to take effect
76_8 (>) Causes full size to take effect

12.12 DCALL

DCALL is used only during initialization to call the display subroutine from segment 2. It is overlayed at runtime by the 6 character DUP name and the pertinent directory information for that DUP.

12.13 TAPDSP

TAPDSP is the routine which runs while trapped tape operations are being processed. It calls the display subroutine, which returns to the AC the number of characters that were displayed. This is added to a counter, TIMER, in segment Ø. If the counter goes positive, the tape has taken too long to process the last command and is assumed to be either in the end zone or otherwise lost.

The remainder of segment 1 is filled with messages, as described in section 13.

13.0 MONITOR Messages

13.1 Messages During Operation of the DEMO Monitor

The following error messages may occur during the operation of the DEMO Monitor.

13.1.1 "SPURIOUS INTERRUPT"

An interrupt has occurred which could not be identified by the DEMO Monitor and either there was no interrupt extension routine, or the interrupt extension routine returned to the location tagged "MARK" rather than "MARK"+ 1 indicating that it could not identify the interrupt.

The interrupt has been cleared by the DEMO Monitor through use of a Power Clear, and processing will continue if a Control C is typed.

13.1.2 "UNCLEARABLE INTERRUPT"

An unidentifiable interrupt has occurred as in 13.1, but Power Clear was unable to clear the interrupt.

In order to proceed, it will be necessary to find the cause of the undesired interrupt and clear it.

13.1.3 "CANNOT FIND
PROGRAM OR GROUP
XXXXXX"

XXXXXX is the six character name entered from the keyboard during the last DEMO INDEX display. No program by that name exists in the DEMO INDEX. If the name is a single letter, this message also indicates that no group exists by that name.

13.1.4 "UNIT n
ILLEGAL WRITE -
BLOCK mmm"

During the execution of a DUP, a Tape Write, Write-Check, or Write-Check-Group instruction has occurred which would write on unit n on block mmm which is outside the range 103-152. It is necessary to restart the current DUP and assure that the program will not attempt to write outside of the assigned area. Restart with Control S or retry the instruction with Control C.

13.1.5 "UNIT n
NOT SELECTED"

An attempt has been made to operate on a tape unit which is not currently in the system. The following problems may have occured:

1. No tape unit has its thumbwheel selector turned to the appropriate unit number.

2. The unit which has its thumbwheel turned appropriately is not in "Remote."

After correcting the problem, typing Control C will effect an attempt to re-execute the same instruction again.

13.1.6 "UNIT n
BAD BLOCK mmm"

The tape control unit is having difficulty moving toward some tape block. The difficulty is occurring near block mmm.

Typing Control C will cause the command to be retried.

13.1.7 "UNIT n
BAD TAPE"

This error message occurs when a tape operation takes unduly long (about four seconds) to complete. Generally, one of the following errors has occurred:

1. A Read-Check or Write-Check instruction cannot be completed because of a checksum error.

2. The tape is in either end-zone and cannot find its way out.

If the latter occurs, the tape can be manually moved out of the end-zone. Typing Control C will cause the command to be re-executed. If you are in end zone, you must manually wind tape.

13.1.8 "UNIT n
SET WRITE ENABLE"

A write instruction to unit n cannot be completed because the tape is protected. Typing Control C will cause the command to be re-executed. Make sure to set the write enable switch.

13.2 Messages Occurring Anytime

The following messages may appear at any time:

13.2.1 "TAPE OPERATION IN PROGRESS"

This message is displayed whenever tape is being moved and there is no more appropriate message to be displayed. During DUP execution, this message is always displayed during tape instruction processing.

13.2.2 "LOADING XXXXXX"

DEMO USER Program XXXXXX is being loaded and will be started as soon as the load is complete.

14.0 Programming Conventions for DEMO User Programs

DEMO User Programs which are to run under the PDP-12 DEMO Monitor are relatively easily implanted in the total DEMO System. Due to certain hardware restrictions it is not possible to check for certain program operations which might interfere with the proper operation of the DEMO Monitor. Accordingly, the following programming restrictions must be placed on DEMO User Programs.

14.1 DEMO User Program Instruction Restrictions

TTY interrupt, ION (General Interrupt), Instruction Trap, and Tape Trap should not be turned off, except as noted in 14.7.1, below.

The DEMO Monitor runs with these PDP-12 features enabled to permit it to check for LINCtape errors and provide overriding control of the TTY keyboard. Disabling of any of the above features will prevent error checking and TTY control and thus should be avoided, although it cannot be prevented.

14.2 Special Functions Register

SF bits 4 Character Size, and 5,Fast Sample,are the only bits in the SF which may be modified by a DEMO Program; otherwise, you run at your own risk.

14.3 DEMO User Program (DUP) Core Use (up to 6K Decimal!)

DUP's are restricted to LINC Memory Segments 2, 3, 4, 5, 6, and 7 (PDP-8 loc. 4ØØØ-17777). PDP-8 page zero references 0-177 are not allowed.

14.4 DUP Start Conditions

DUPs must start in LINC mode at Loc. 2Ø in segment 2 although they may contain either PDP-8, LINC, or mixed mode.

14.5 DUP use of LINCtape (SCRATCH.)

LINCtape blocks 0-2Ø unit Ø and 16-17 unit 1 are reserved for use by the DEMO Monitor, since DEMO Monitor tapes also contain DIAL, the LINCtape blocks used by DIAL are also reserved. Blocks 103-152 are reserved for a scratch area (SCRATCH.) and is the only legal area DEMO Monitor recognizes as writeable by DEMO User Programs.

14.6 DUP Write Protection

LINCtape blocks 103-152 on any unit should be available for writing. Reading is permitted anywhere on any unit. SCRATCH blocks 103-152 is a DIAL file which reserves those blocks on DEMO Monitor tapes. Attempts to write on any other blocks of tape are stopped by the DEMO Monitor. Control C retries the illegal instruction; Control S reloads the DUP.

14.7 Tape Extended Operations Register

TEOR bits 1Ø and 11 (extended unit bits) are the only TEOR bits which may be altered by a DUP.

14.7.1 NO PAUSE Tape Operations

Tape operation may only be performed in NO PAUSE mode by disabling tape traps before executing the tape instruction. Operations performed in this manner have full use of the Tape Extended Operations Register.

It should be noted that tape operations performed with tape trap turned off are not checked at all by the DEMO Monitor.

14.8 Operate Instruction, OPR Usage (also see paragraph 11.2)

Several OPR instructions are provided for use by DUPs. The following paragraph specifies these instructions and their use.

14.8.1 RETV (OPR5)

RETV may be used to retrieve tape location information about any file in the DEMO Index. It is called as follows:

14.8.1 RETV (OPR5) (continued)

```
RETV                    /OPR5
ADDR                    /Address of parameter region
 .
 .
 .
```

ADDR must be a LINC-mode address. The parameter region must be in the current Instruction Field. Its format is:

```
ADDR,  TEXT         ".FILENAM"
       5606
       1114
       0516
       0115
```

.FILENAM must be the full 8 character name as it appears in the DIAL Index and if not 8 characters, fill with 77_8.

RETV returns the following information in ADDR+4, 5, 6, and 7:

ADDR+4	Source block number
ADDR+5	Source length
ADDR+6	Binary block number
ADDR+7	Binary length

If no entry is found in the DEMO INDEX for .FILENAM, ADDR+4, 5, 6, and 7 will not be altered.

14.8.2 NEXT (OPR6)

NEXT is a variation of EXIT (14.8.3) which causes the next DUP which is a member of the same group to be loaded in preparation for execution. It is logically equivalent to typing Control N during DUP execution.

If there are no further DUPs in the group, Display Index will be entered.

14.8.3 EXIT (OPR7)

EXIT is provided to be used in the place of either PDP-8 mode or LINC-mode HLT instructions, although it must be executed in LINC-mode. It will redisplay the DEMO Index.

14.8.4 XCHN (OPR11)

XCHN extends the interrupt service chain to allow processing of interrupts from devices other than the console Teletype or LINCtape. Its use is as follows:

```
    XCHN                    /OPR11
PMODE
    RNAME                   /Routine name in full PDP-8 mode
LMODE                        Address
    .
    .
    .
```

The following restrictions apply to the routine RNAME:

Interrupts must not be enabled.

The routine must start and finish in PDP-8 mode although LINC-mode may be used within the routine.

The routine should be in the following format:

```
RNAME, Ø                    /Entry point
  .
  .                         /Test devices and
  .                         /set or clear
  .                         / internal flags

JMP I RNAME                 /Return - no flags found
ISZ   RNAME                 /Return to .+1 if the
JMP I RNAME                 /source of the interrupt
                            /was identified.
```

14.8.5 OKST (OPR12)

OKST is provided to perform the same function as KST. Because the DEMO Monitor operates with interrupts on, the keyboard flag will never be apparent to the DUP. OKST tests an internal keyboard flag and functions just like KST. Note that OKST can not be used indirectly, although KST can (KST I).

14.8.6 KBDA (OPR13)

KBDA returns the full ASCII keyboard character last received in the AC. The character is echoed, and line feeds are provided for carriage returns. If no character has been received since the last KBD or KBDA, then one of the following occurs:

If the I bit is on in the KBDA instruction, the DEMO Monitor waits for a character from the keyboard before returning.

If the I bit is not on, the instruction returns with a clear AC.

14.8.7 TYP (OPR14)

TYP causes the full ASCII character in the AC to be typed on the console TTY.

14.8.8 KBD (OPR15)

KBD functions similarly to KBDA, except that the LINC-code is returned in the AC rather than the full ASCII code. If the code returned is a CASE (23_8), the next KBD will return the LINC-code.

If no character has been received since the last KBDA, the instruction behaves the same as KBDA (see paragraph 14.8.6).

14.9 DUP Instruction Restrictions

The DUP should not halt or do anything which leaves the system in limbo (i.e., IOF; JMP .). These situations cannot be trapped or detected by the DEMO Monitor.

15.0 DUP TEXT Files (Summary and Help Frames)

DUP Help and Summary Frame files are prepared using the TEXT pseudo-op in DIAL. It should be noted that the DEMO Monitor expects to find an assembled binary file without a header block located in the blocks indicated in the DEMO INDEX as the source blocks of a DUP. We do not use the source blocks as actual source, but rather binary DUP TEXT files containing the Summary and Help Text for that DUP. They follow the standard PDP-12 QANDA conventions, plus the followng rules:

1. DUP display frames cannot exceed 256 characters (including QANDA control characters).

2. DUP display files must contain the Summary Frame as the first frame of the file.

3. DUP display files must contain an odd number of DEMO Help Frames (Summary Frame plus odd number of Help Frames even number of tape blocks, as illustrated in figure 15-2.

4. DUP display frames must be aligned on 200_8 word boundaries beginning at *Ø.

15.1 DUP Summary and Help Frame Assembly and Storage Procedure

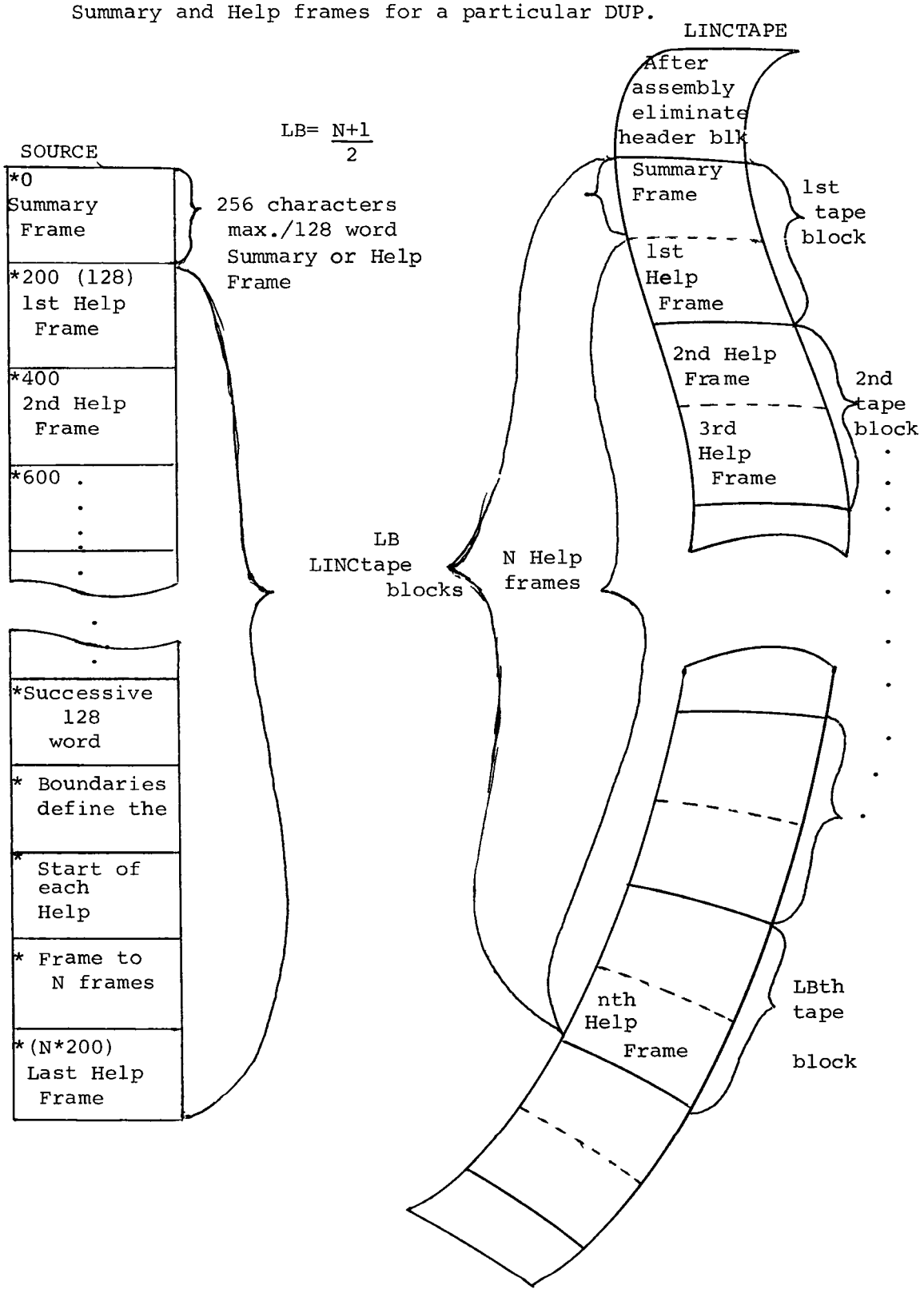
1. Assemble as usual.

2. Store as some scratch binary file, (i.e., SB DOG, N).

3. Create a fake source file for the DUP for which the current text is being prepared which is one block less than the number of blocks in the saved binary of that text.

4. Using the PI procedure, transfer the saved binary text (with the header block ignored) to the fake source file just created.

N = Number of Help Frames (must always be odd)
LB = Number of LINCtape blocks required to keep
Summary and Help frames for a particular DUP.



16.0 Creating a DEMO Monitor Tape

The following procedure may be used to create a DEMO Monitor tape after any changes or modifications have been made to any Monitor Code.

16.1 Cross Segment References

Each subsystem of the DEMO Monitor (except LOADER) has a section of references to other subsystems. In the event that a modification is made which moves any of the referenced locations, the reference assignments in other subsystems must be revised.

16.2 Assembly

Each subsystem is self-contained and may be assembled as an entity in the conventional manner using DIAL, saving binaries of each subsystem.

16.3 The DEMO Monitor binaries

The binaries should be copied as noted below to create a self-loading DEMO Monitor. See figure page 16-2.

Note that the first block of the saved binary is a header block, so the first program block is the one after the header block from the DIAL Directory. The last block in the binary of SEGØ and SEG1 is empty and automatically generated by DIAL Assembly. See section 4.4 of the LAP6-DIAL Manual, DEC-12-SEZB-D.

16.3.1 "PIPing" the pieces together

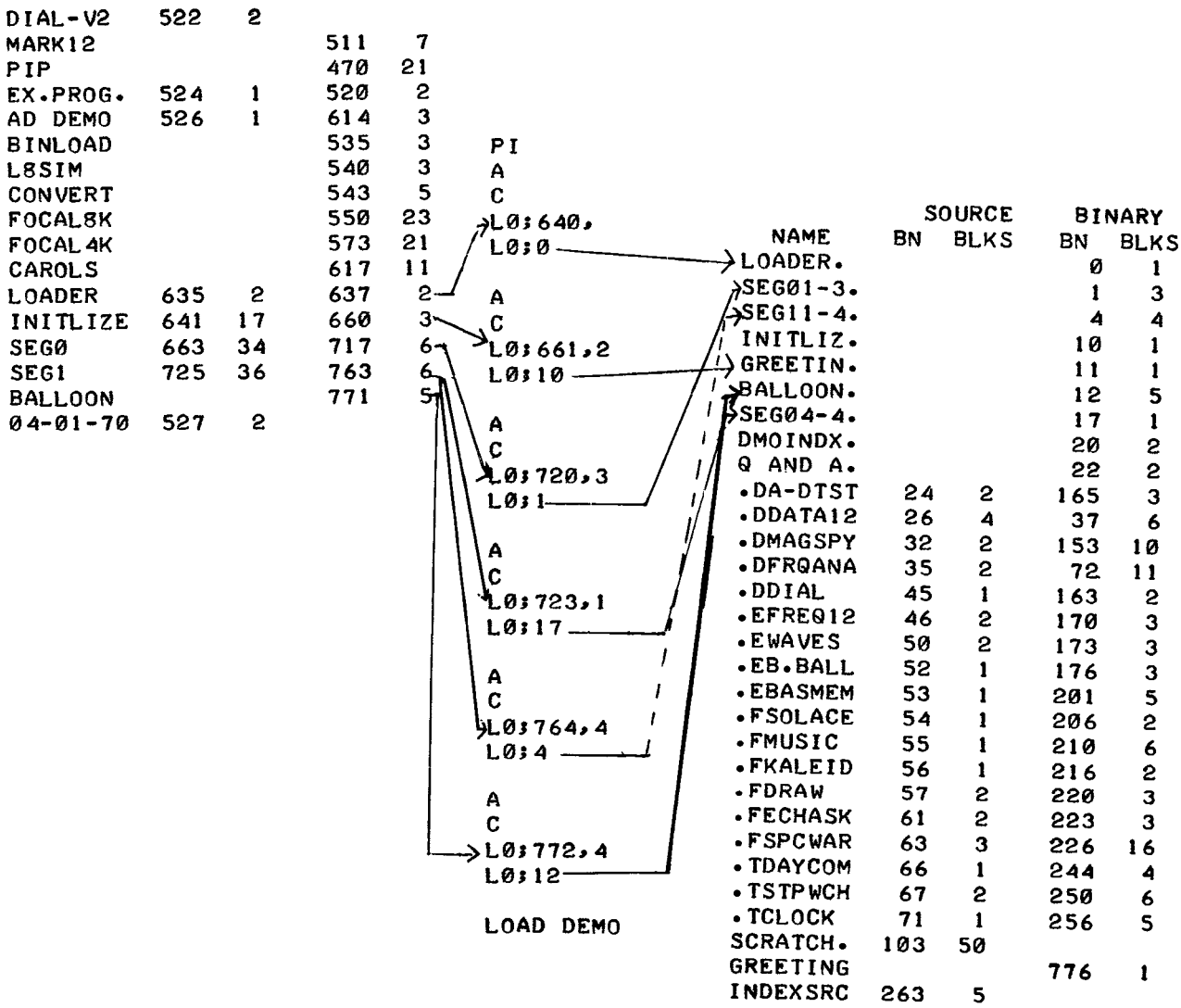
FILE	RELATIVE FILE:	
SEGØ	Block 2	Copy to DEMO Monitor Tape Block 1
SEGØ	Block 3	Copy to DEMO Monitor Tape Block 2
SEGØ	Block 4	Copy to DEMO Monitor Tape Block 3
SEGØ	Block 5	Copy to DEMO Monitor Tape Block 17
SEG1	Block 2	Copy to DEMO Monitor Tape Block 4
SEG1	Block 3	Copy to DEMO Monitor Tape Block 5
SEG1	Block 4	Copy to DEMO Monitor Tape Block 6
SEG1	Block 5	Copy to DEMO Monitor Tape Block 7
INITLIZE	Block 2	Copy to DEMO Monitor Tape Block 1Ø
INITLIZE	Block 3	Copy to DEMO Monitor Tape Block 11
LOADER	Block 2	Copy to DEMO Monitor Tape Block Ø

16.4 Talk BALLOON

There is no symbolic for the Talk BALLOON. The binary for the talk balloon is BALLOON.

FILE:		RELATIVE FILE:	
BALLOON	Block 2	Copy to DEMO Monitor Tape	Block 12
BALLOON	Block 3	Copy to DEMO Monitor Tape	Block 13
BALLOON	Block 4	Copy to DEMO Monitor Tape	Block 14
BALLOON	Block 5	Copy to DEMO Monitor Tape	Block 15

16.5 DEMO INDEX Creation



17.0 Operating Procedures (see Flow Chart next page)

17.1 Loading - Two tapes (DEMO12 tapes only)

1. Mount both tapes on units and set the units to Ø and 1. Because the tapes are identical, either tape can be on either unit.

2. Write Enable both tapes.

3. Place Ø7ØØ in the Left Switches and ØØØØ in the Right Switches.

4. Stop the computer if it is running.

5. Press I/O PRESET

6. Press DO and wait for tape to stop moving

7. Press START 2Ø

The system will load and begin execution. No further operations are required except as requested on the display scope.

17.2 Loading - One Tape

In the event that a special tape is to be used with a DUP it will be required that the Index and Display Index be written on tape Ø instead of 1. The startup procedure is then the same as for two tapes, but the DEMO Monitor tape must be on unit Ø (Write Enabled), and tape 1 must not be Write Enabled. In order for DEMO Monitor to recognize unit 1 for its Index, use initialization must take place again. Likewise, if initialization starts with unit 1 on, the Monitor assumes it should be on during all your DEMO use. The only way to get DEMO to stop looking at unit 1 is to go through initialization again.

17.3 Restarting

If it is necessary to restart the DEMO Monitor, the starting procedure (including initialization) must be reexecuted. The loading procedure in 17.1 or 17.2 should be followed or else use of the DIAL File AD DEMO will start the DEMO Monitor without having to toggle the console switches.

17.4 Terminating

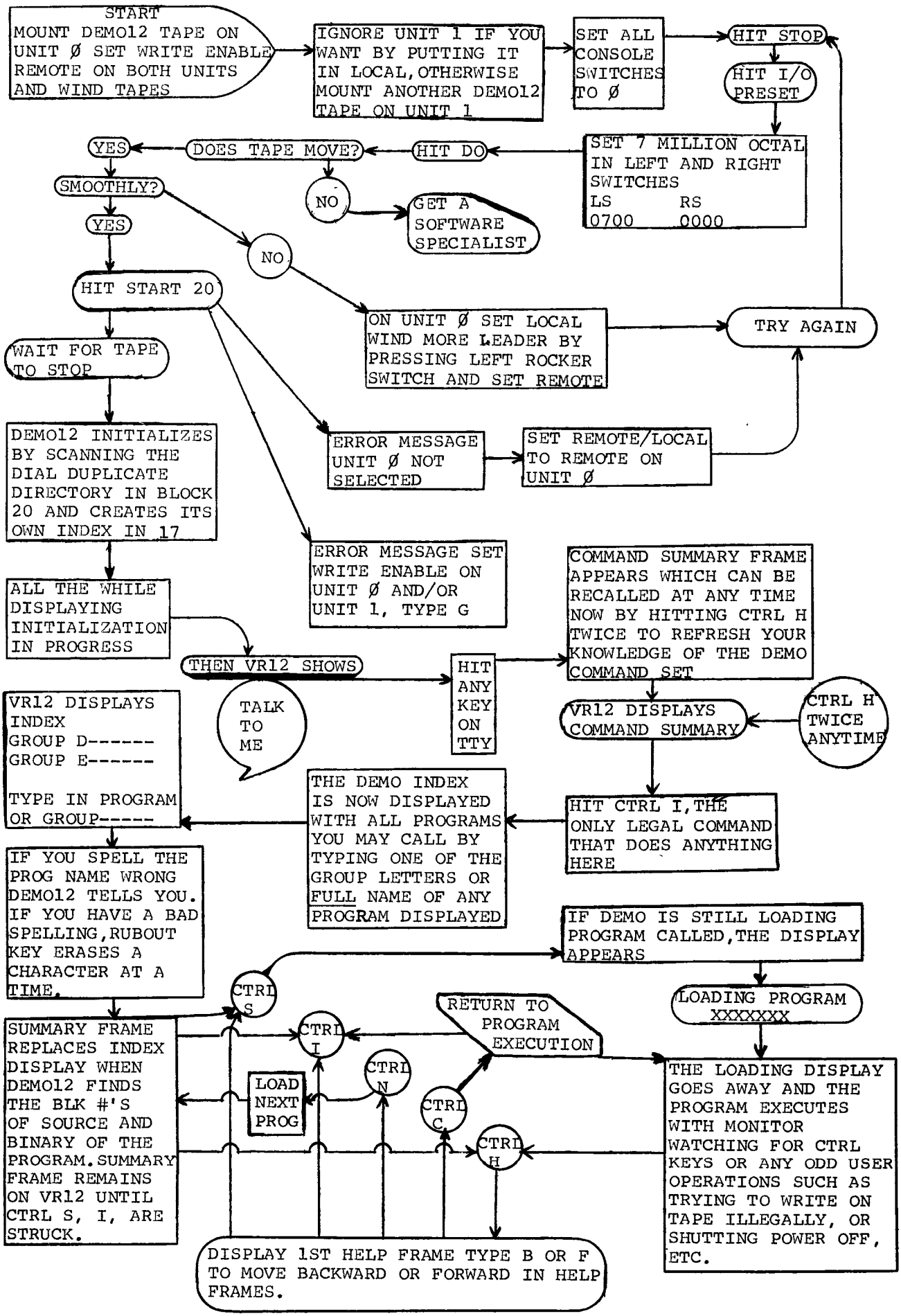
No special procedure is required for terminating the DEMO Monitor.

17.5 Error Conditions

All runtime errors are handled similarly. In all cases, a reasonable state will be entered if a Control C is struck except in the case where an unclearable interrupt occurs. In that case, a restart is required after the unclearable interrupt is removed.

If the initialization routines have any difficulty reading in the Block 20 Index, the message "BAD TAPE - RESTART DEMO" may appear. Occasionally, a simple restart from step 2 of 17.1 will get past the error. If that fails, tape units Ø and 1 should be swapped and an attempt should be made to run from the other tape.

DEMO12 USE FLOW CHART



DIAL Character Set

Keyboard	External (ASCII)	Internal
A	301	1
B	302	2
C	303	3
D	304	4
E	305	5
F	306	6
G	307	7
H	310	10
I	311	11
J	312	12
K	313	13
L	314	14
M	315	15
N	316	16
O	317	17
P	320	20
Q	321	21
R	322	22
S	232	23
T	324	24
U	325	25
V	326	26
W	327	27
X	330	30
Y	331	31
Z	332	32
[(SHIFT/K)	333	33
\ (SHIFT/L)	334	34
] (SHIFT/M)	335	35
↑	336	36
←	337	Illegal (not displayed)
SPACE	240	40
!	241	41
"	242	42
#	243	Illegal (not displayed)
$	244	44
%	245	45
&	246	46
'	247	Illegal (not displayed)
(	250	50
)	251	51
*	252	52

Character Set (Cont)

Keyboard	External (ASCII)	Internal
+	253	53
,	254	54
-	255	55
.	256	56
/	257	57
ø	260	60
1	261	61
2	262	62
3	263	63
4	264	64
5	265	65
6	266	66
7	267	67
8	270	70
9	271	71
:	272	72
;	273	73
<	274	74
=	275	75
>	276	76
?	277	77
@	300	Illegal (not displayed)
LINE FEED	212	37
RETURN	215	43 (not displayed)
ALTMODE	375	None " "
RUBOUT	377	None " "
CONTROL/I (TAB)	211	47 " "

HOW TO OBTAIN SOFTWARE INFORMATION

Announcements for new and revised software, as well as programming notes, software problems, and documentation corrections are published by Software Information Service in the following newsletters.

Digital Software News for the PDP-8/12 Family
Digital Software News for the PDP-9/15 Family
PDP-6/PDP-10 Software Bulletin

These newsletters contain information applicable to software available from Digital's Program Library.

Please complete the card below to place your name on the newsletter mailing list.

Questions or problems concerning DEC Software should be reported to the Software Specialist at your nearest DEC regional or district sales office. In cases where no Software Specialist is available, please send a Software Trouble Report form with details of the problem to:

Software Information Service
Digital Equipment Corporation
146 Main Street, Bldg. 3-5
Maynard, Massachusetts 01754

These forms, which are available without charge from the Program Library, should be fully filled out and accompanied by teletype output as well as listings or tapes of the user program to facilitate a complete investigation. An answer will be sent to the individual and appropriate topics of general interest will be printed in the newsletter.

New and revised software and manuals, Software Trouble Report forms, and cumulative Software Manual Updates are available from the Program Library. When ordering, include the document number and a brief description of the program or manual requested. Revisions of programs and documents will be announced in the newsletters and a price list will be included twice yearly. Direct all inquiries and requests to:

Program Library
Digital Equipment Corporation
146 Main Street, Bldg. 3-5
Maynard, Massachusetts 01754

Digital Equipment Computer Users Society (DECUS) maintains a user Library and publishes a catalog of programs as well as the DECUSCOPE magazine for its members and non-members who request it. For further information please write to:

DECUS
Digital Equipment Corporation
146 Main Street
Maynard, Massachusetts 01754

Send Digital's software newsletters to:

Name_____
Company Name_____
Address_____
_____
(zip code)

My computer is a

PDP-8/I ☐		PDP-8/L ☐	
LINC-8 ☐		PDP-12 ☐	
PDP-9 ☐		PDP-15 ☐	Please specify
PDP-10 ☐		OTHER ☐	_____

My system serial number is_____ (if known)

Fold Here

Do Not Tear - Fold Here and Staple

FIRST CLASS
PERMIT NO 33
MAYNARD, MASS.

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

Postage will be paid by:

digital

Digital Equipment Corporation
Software Information Services
146 Main Street, Bldg. 3-5
Maynard, Massachusetts 01754

READER'S COMMENTS

DEMO12 TECHNICAL REFERENCE MANUAL
DEC-12-MRZA-D

Digital Equipment Corporation maintains a continuous effort to improve the quality and usefulness of its publications. To do this effectively we need user feedback – your critical evaluation of this manual.

Please comment on this manual's completeness, accuracy, organization, usability, and readability.

_____

_____

_____

_____

_____

Did you find errors in this manual? _____

_____

_____

_____

_____

How can this manual be improved? _____

_____

_____

_____

_____

_____

DEC also strives to keep its customers informed of current DEC software and publications. Thus, the following periodically distributed publications are available upon request. Please check the appropriate boxes for a current issue of the publication(s) desired.

☐ Software Manual Update, a quarterly collection of revisions to current software manuals.

☐ User's Bookshelf, a bibliography of current software manuals.

☐ Program Library Price List, a list of currently available software programs and manuals.

Please describe your position. _____

Name _____ Organization _____

Street _____ Department _____

City _____ State _____ Zip or Country _____

— — — — — — — — — — — — — — — — — — Fold Here —

— — — — — — — — — — — — — — — — Do Not Tear - Fold Here and Staple — — — — — — — — — — — —

FIRST CLASS
PERMIT NO. 33
MAYNARD, MASS.

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

Postage will be paid by:

digital

Digital Equipment Corporation
Software Information Services
146 Main Street, Bldg. 3-5
Maynard, Massachusetts 01754